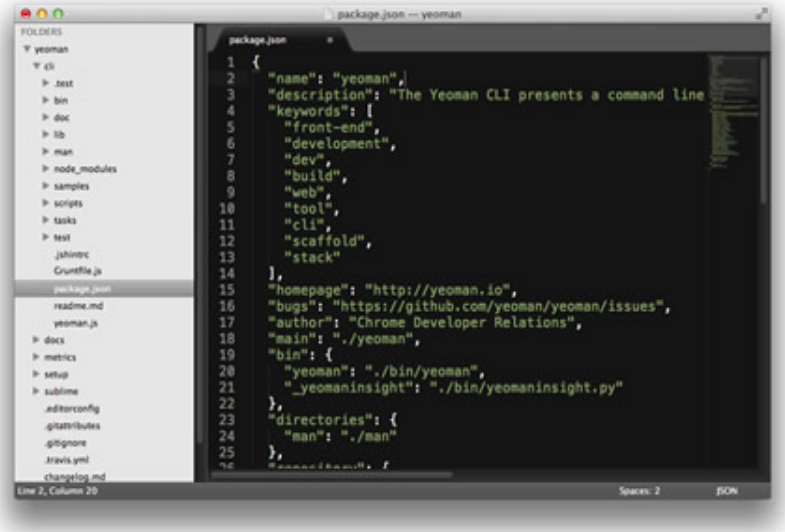
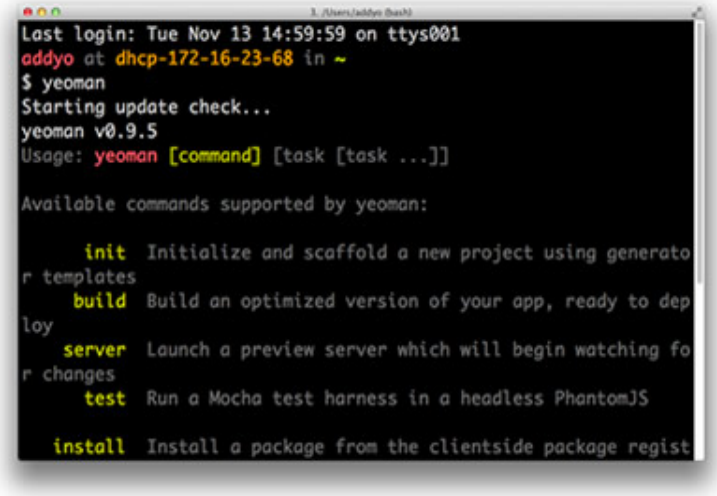
TOOLING FOR THE MODERN WEBAPP DEVELOPER

@addyosmani

Browser + DevTools



Text Editor

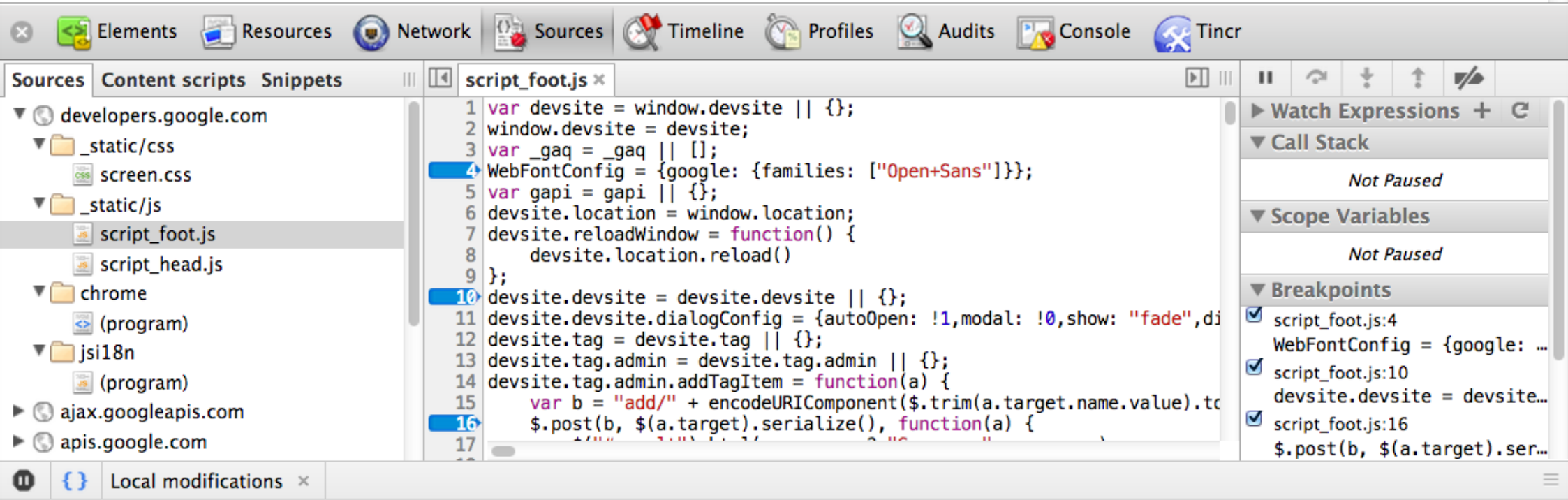


Terminal

Devices

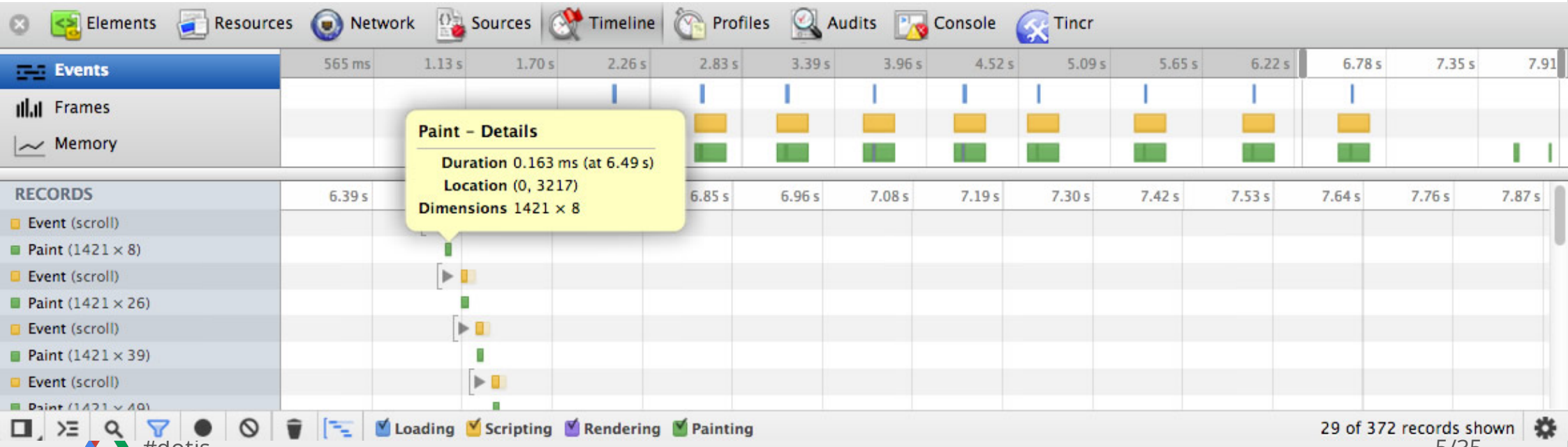
In-browser DevTools

- Constantly evolving
- Use Canary channel for development
- Lots of juicy experimental features



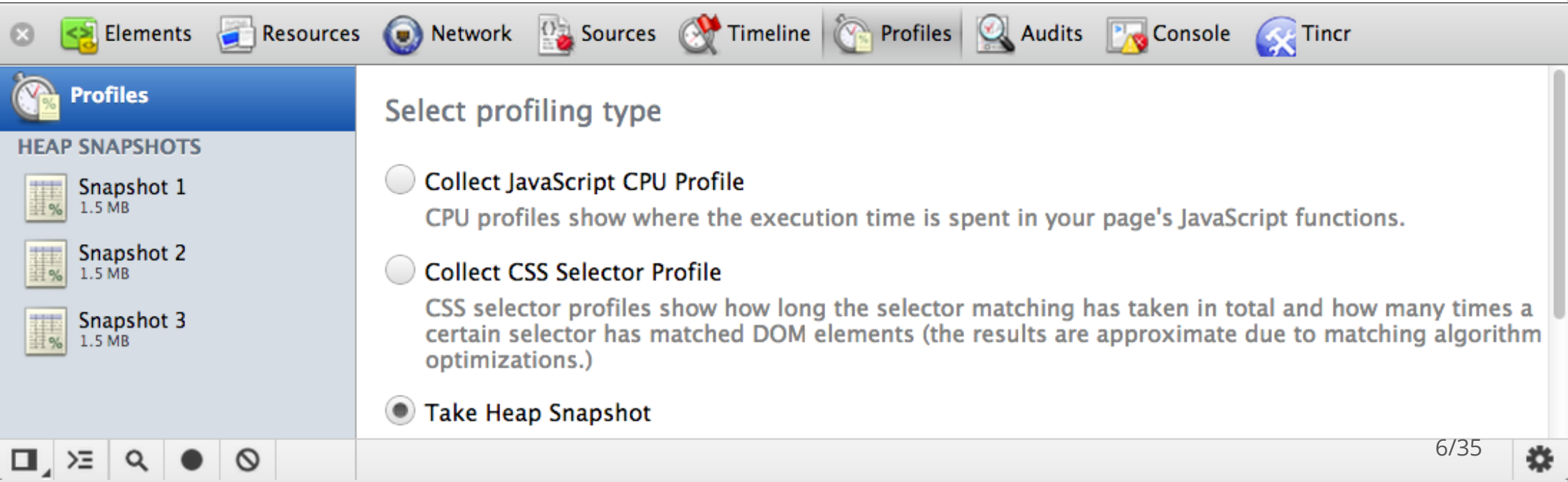
Performance: Timeline + Frames view

- Timeline gives you an overview of memory usage over time
- Summary and detailed views
- Helps remove jank. Layout or scripts - who triggered what?
- Frames view helps achieve that snappy 60fps you ideally want



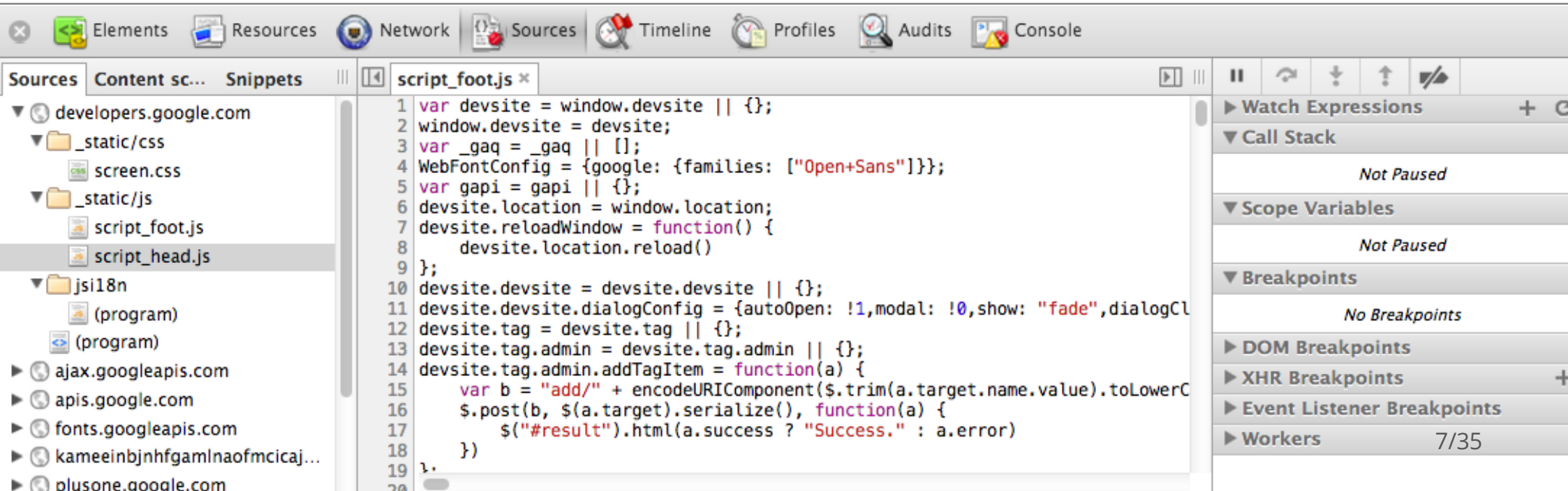
Finding memory leaks and DOM leaks

- JavaScript, CSS, Heap snapshot Profiles
- What is using memory at a given point in time? Not being GC'd?
- Use comparison view to identify potential memory leaks
- Use summary view to identify DOM leaks



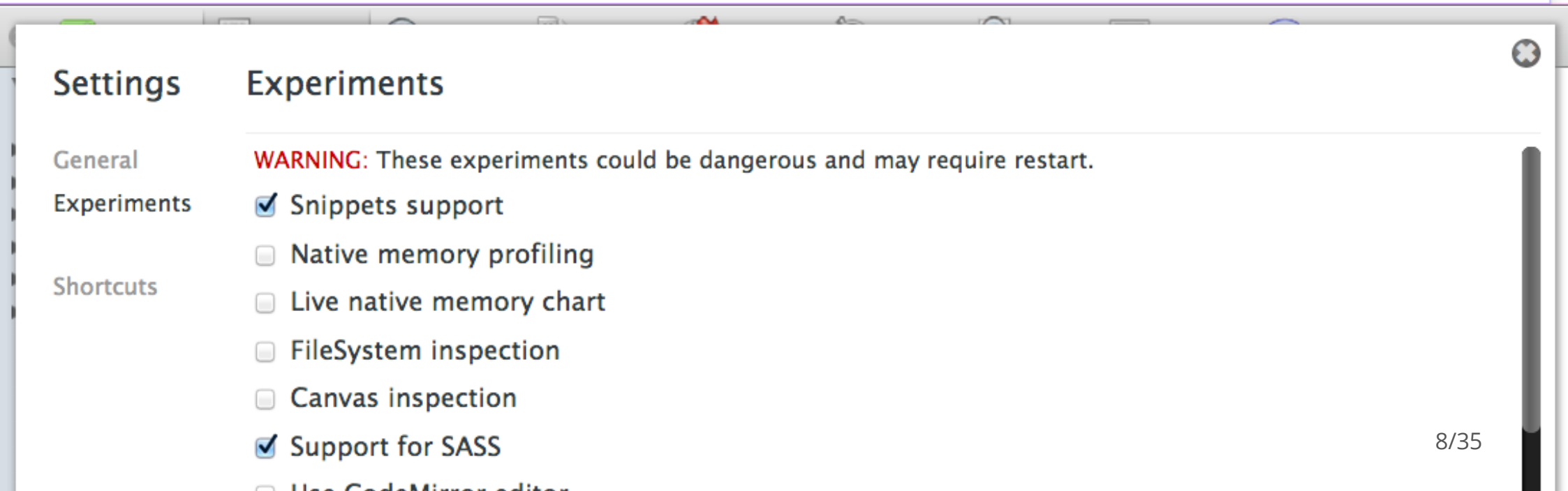
A better authoring workflow

- Live Edit
- Snippets
- Revision history
- AutoSave



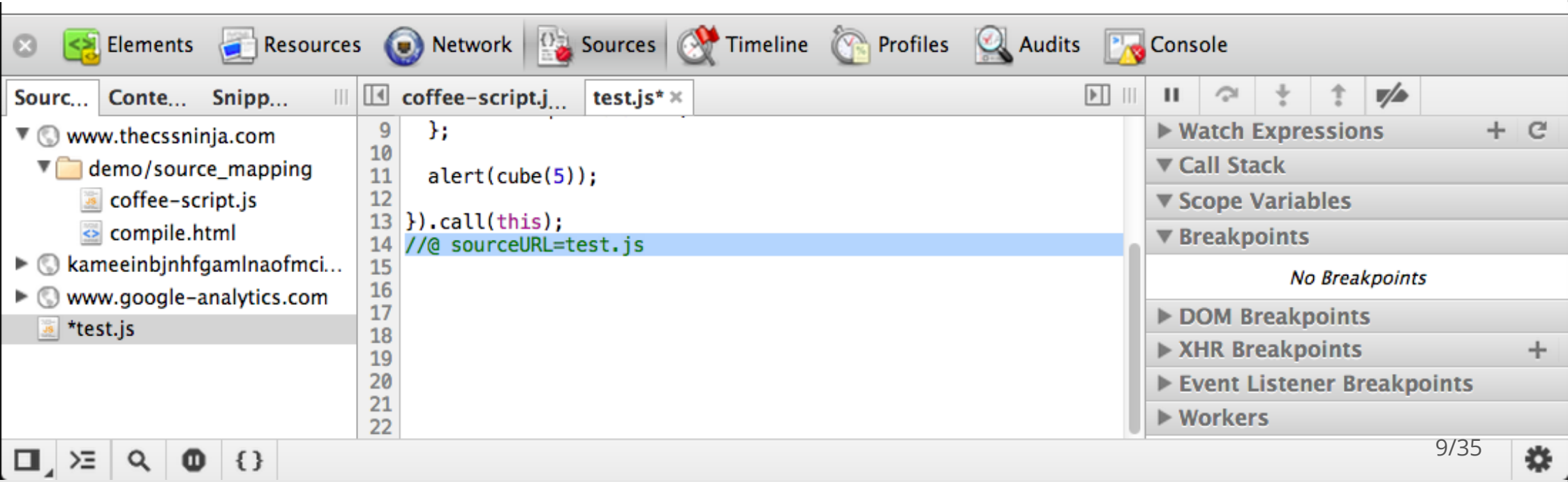
Live reloading + SASS Source Maps

- Editing compiled CSS has little value
- Enable new SASS hotness
- BOOM! You can edit SASS source files
- Changes automatically reload



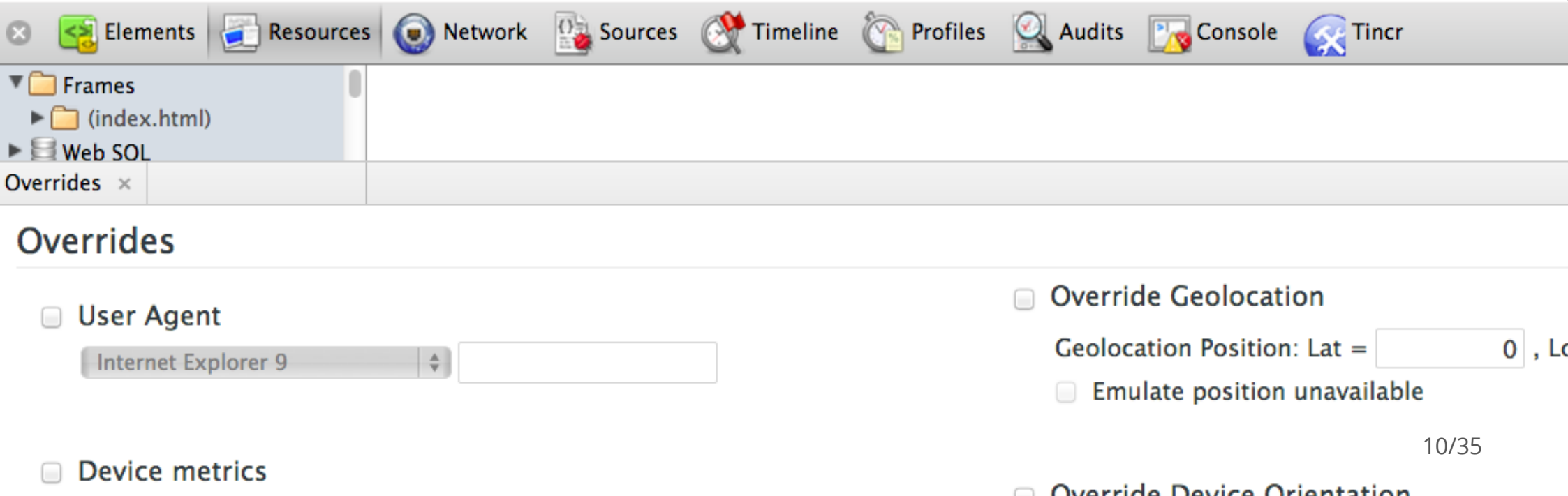
CoffeeScript + @sourceURL

- Compile your Coffee sources
- Open up the DevTools
- Review your compiled file
- Whoa! sourceURL comments



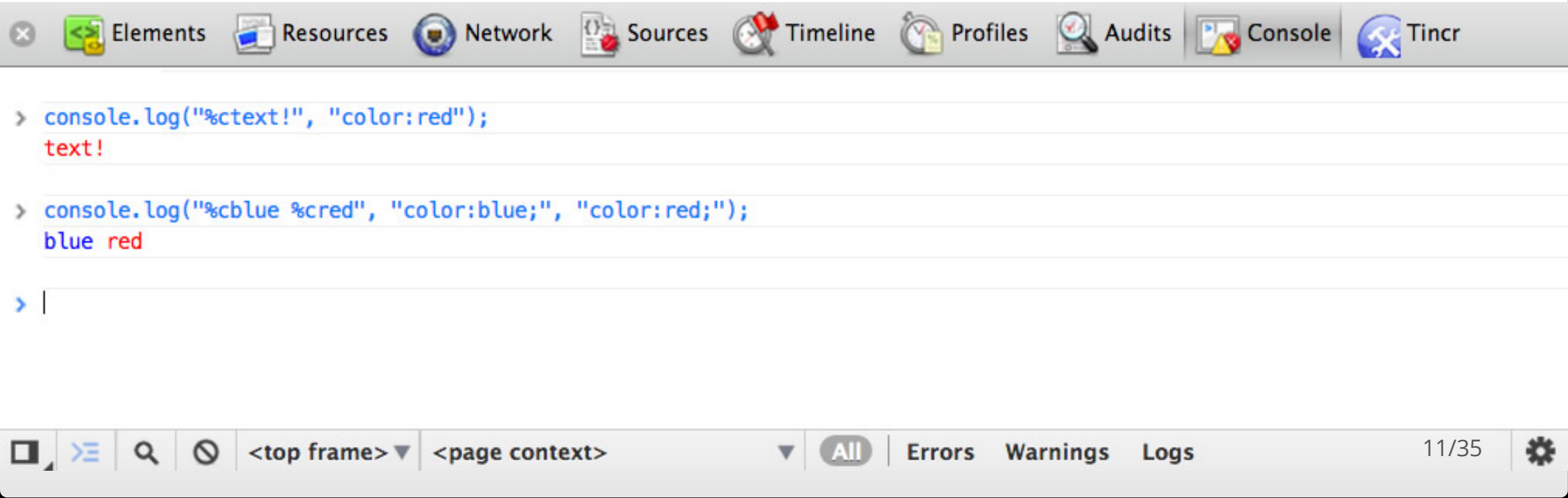
Mobile Debugging: Overrides Panel

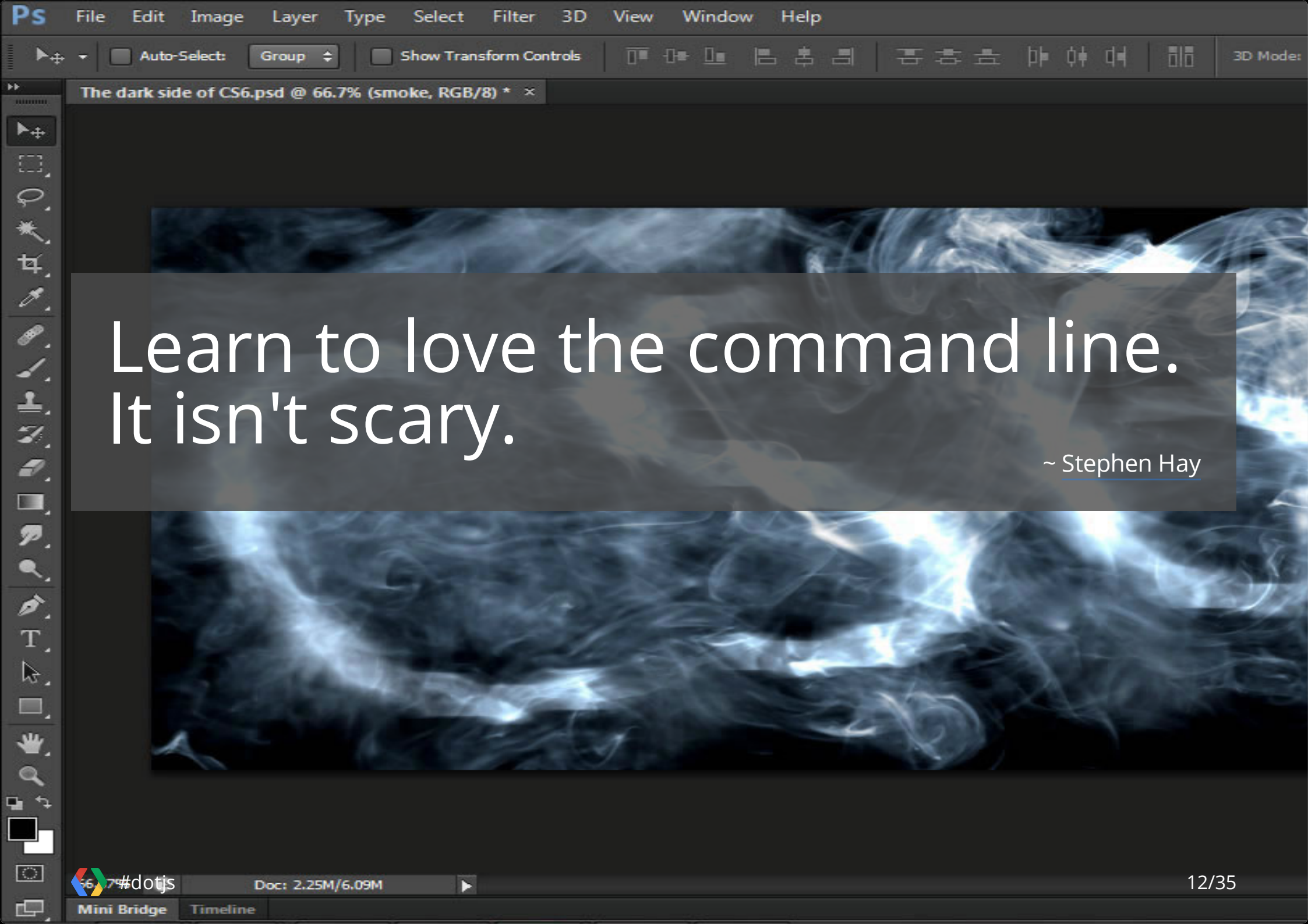
- User Agent
- Device Metrics
- Geolocation
- Orientation
- Emulate touch events



DevTools Console

- Styled-console
- Multi-style support
- inspect() command



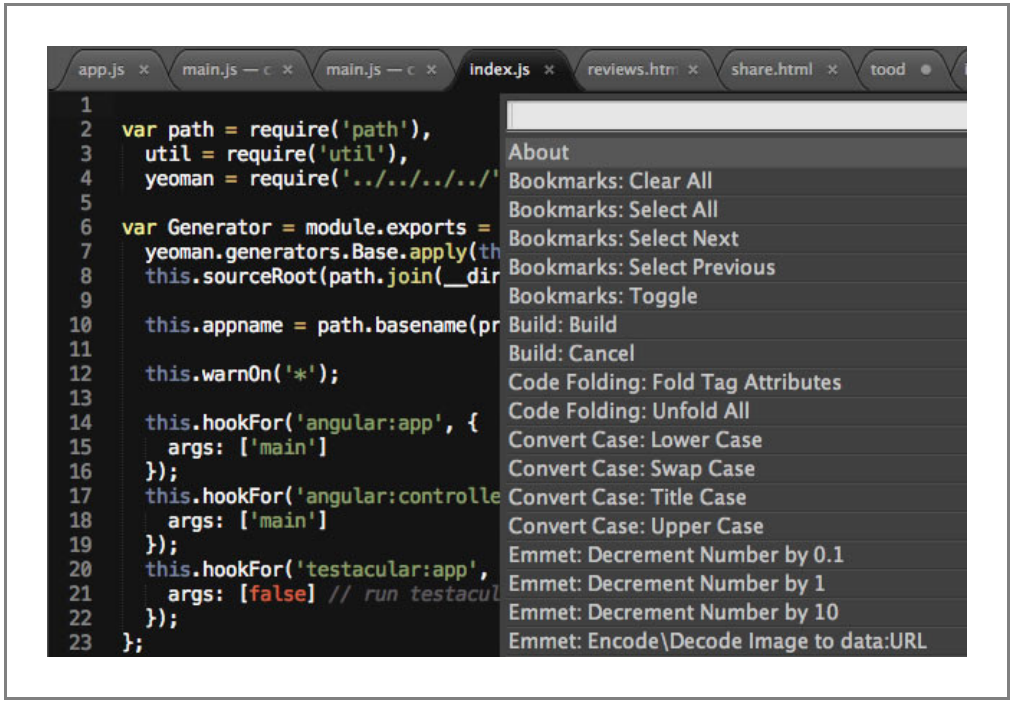

Ps
File Edit Image Layer Type Select Filter 3D View Window Help
Auto-Select: Group Show Transform Controls
3D Mode:
The dark side of CS6.psd @ 66.7% (smoke, RGB/8) *
Learn to love the command line.
It isn't scary.
~ Stephen Hay
Doc: 2.25M/6.09M
Mini Bridge
Timeline

Command-line

- Make it look hot ♡.
- Capture and replay your command line *history*
- DOTFILE EVERYTHING (mine)
- Aliases are awesome
- Our faves: `gz` , `server` alias and `ny`
- Some for Browserstack: win7ie8, win8ie10, ios3

Sublime Text #protips

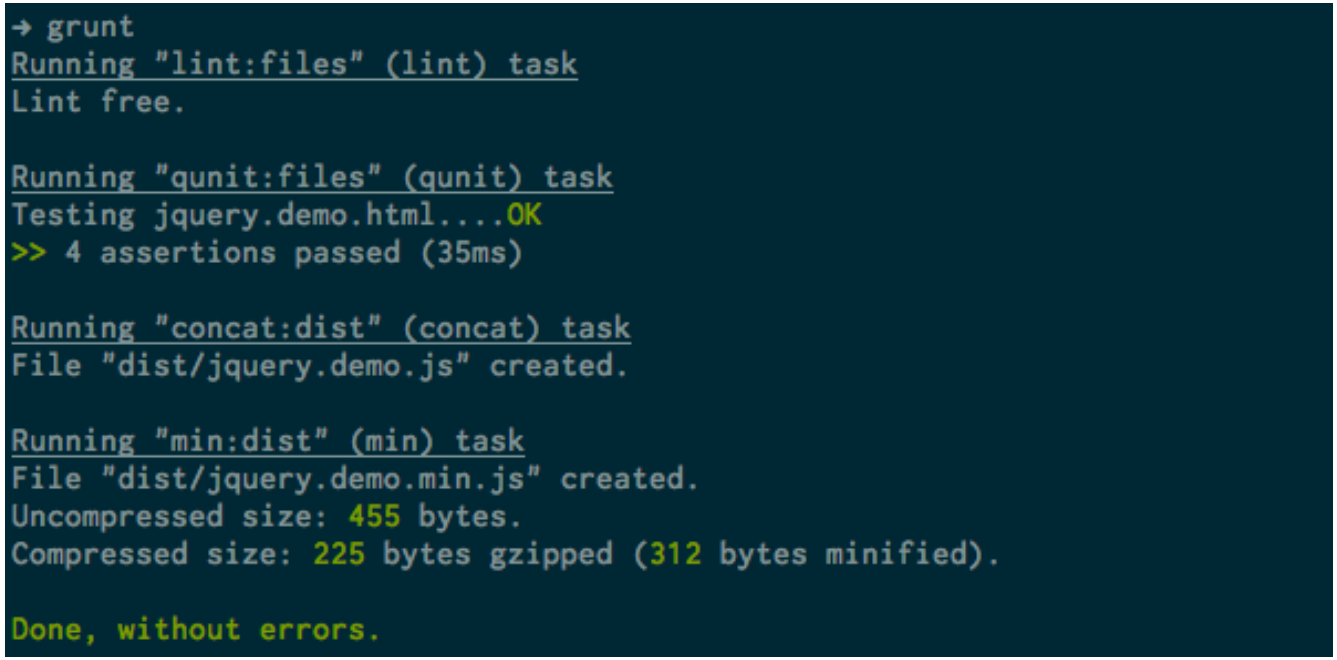
- Run JavaScript From Your Editor In The Browser
- Use the built-in Build System!
- Zen coding with Emmet
- Stack Overflow code search
- Snippets for frameworks




Finding missing semicolons should never be a manual process.
write less, do more.
THANKS OUR GENEROUS SPONSORS
(mt) mediatemple
Microsoft .NET
ASP.NET

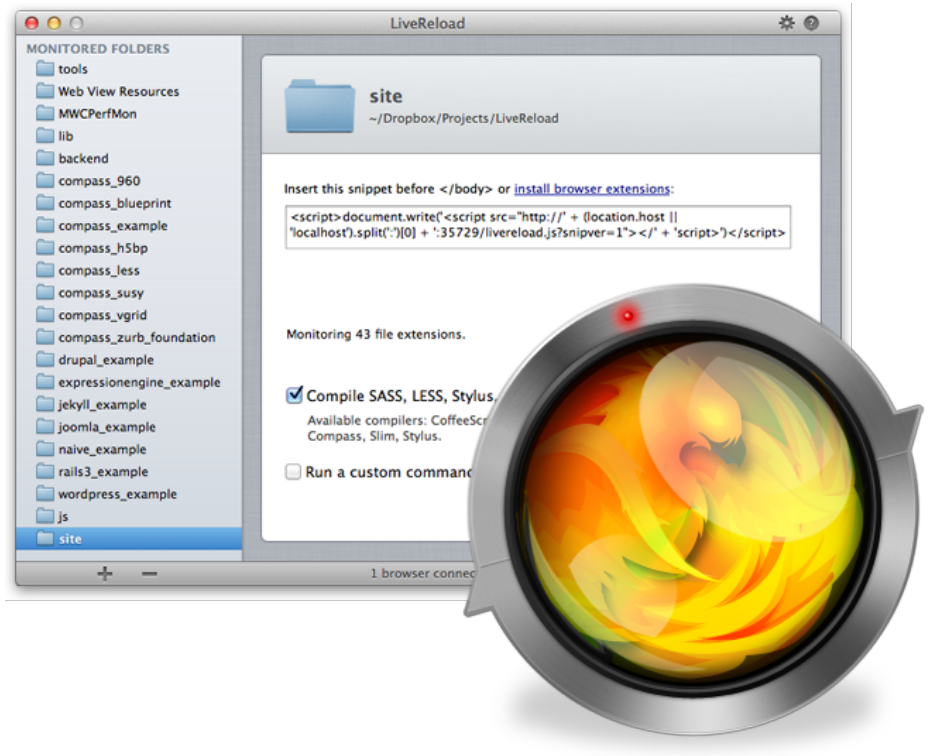
Linting

- On file save
- On source control commit
- At build time
- *Anything better?*



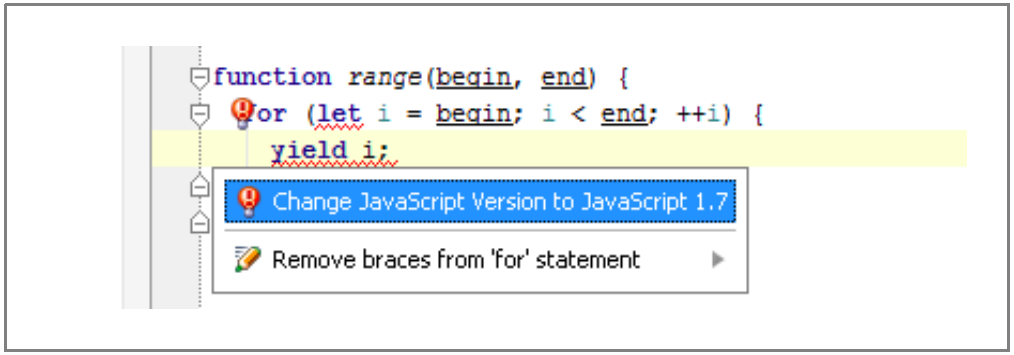
Live feedback

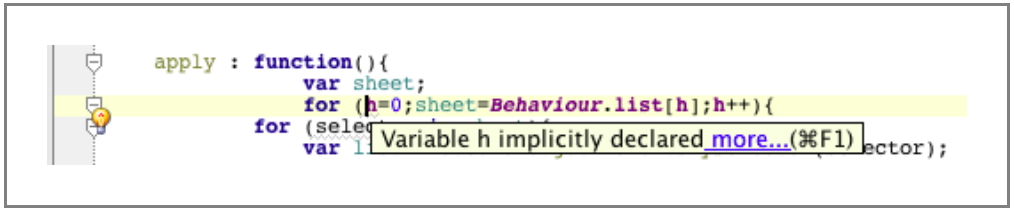
- Linting
- Reload
- Recompilation

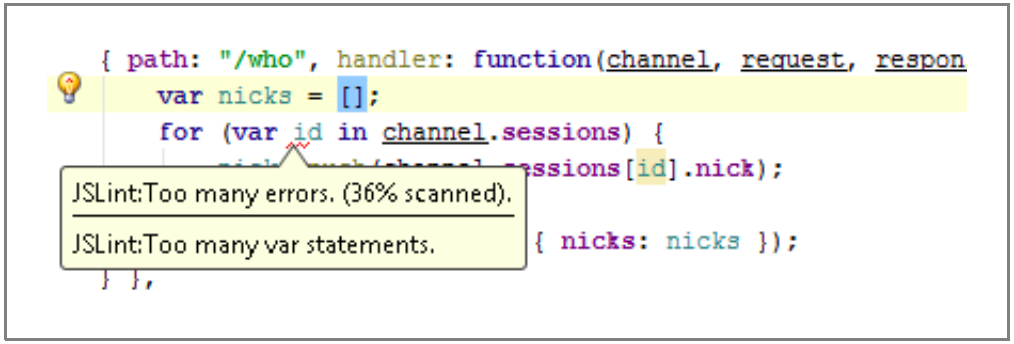


WebStorm

- Live Edit + Chrome
- JS Language version based suggestions
- Code inspection and zen coding
- Suggestions for DRYer code
- Built in code linting
- and more







Unit Testing In The Cloud

Testing approaches you already know:

- In the browser
- In a headless browser on-demand via cmd line: `grunt qunit`
- In a headless browser post-push

New hotness:

- In multiple browsers *in the cloud* via cmd line:

```
bunyip  -f Modernizr/test/index.html -c ~/bunyip/config.js -b ios
```

```
bunyip -f index.html local -l "firefox|chrome|safari|phantomjs"
```

Build system

Lint. Resolve depenencies. concatenate modules. compile. Flatten your CSS `@imports`. Remove debugging statements. Compress images. Precompile templates. Run tests in a variety of environments. Revs asset paths for caching. Affirm code quality.

New hotness: Grunt.js

GRU

Grunt is a task-
JavaScript proj

View on GitHub | Docu

Getting started

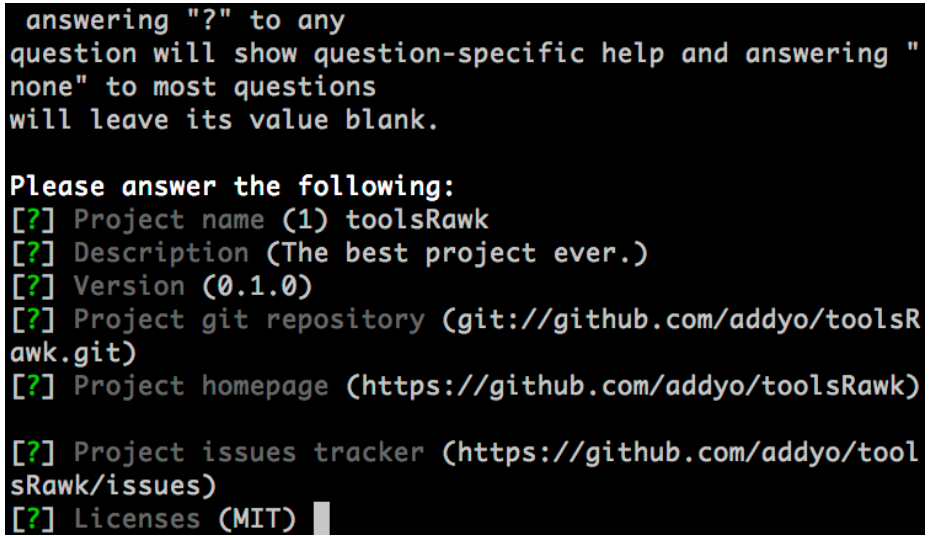
Be sure to read the getting started guide, which is a complete guide to configuring grunt for your project. In addition, check out the example gruntfiles which highlight a number of fairly common configurations.

Latest plu

contrib-ember Manage and
contrib-manifest Precompile
pistachio-compiler Grunt task t

Grunt.js:

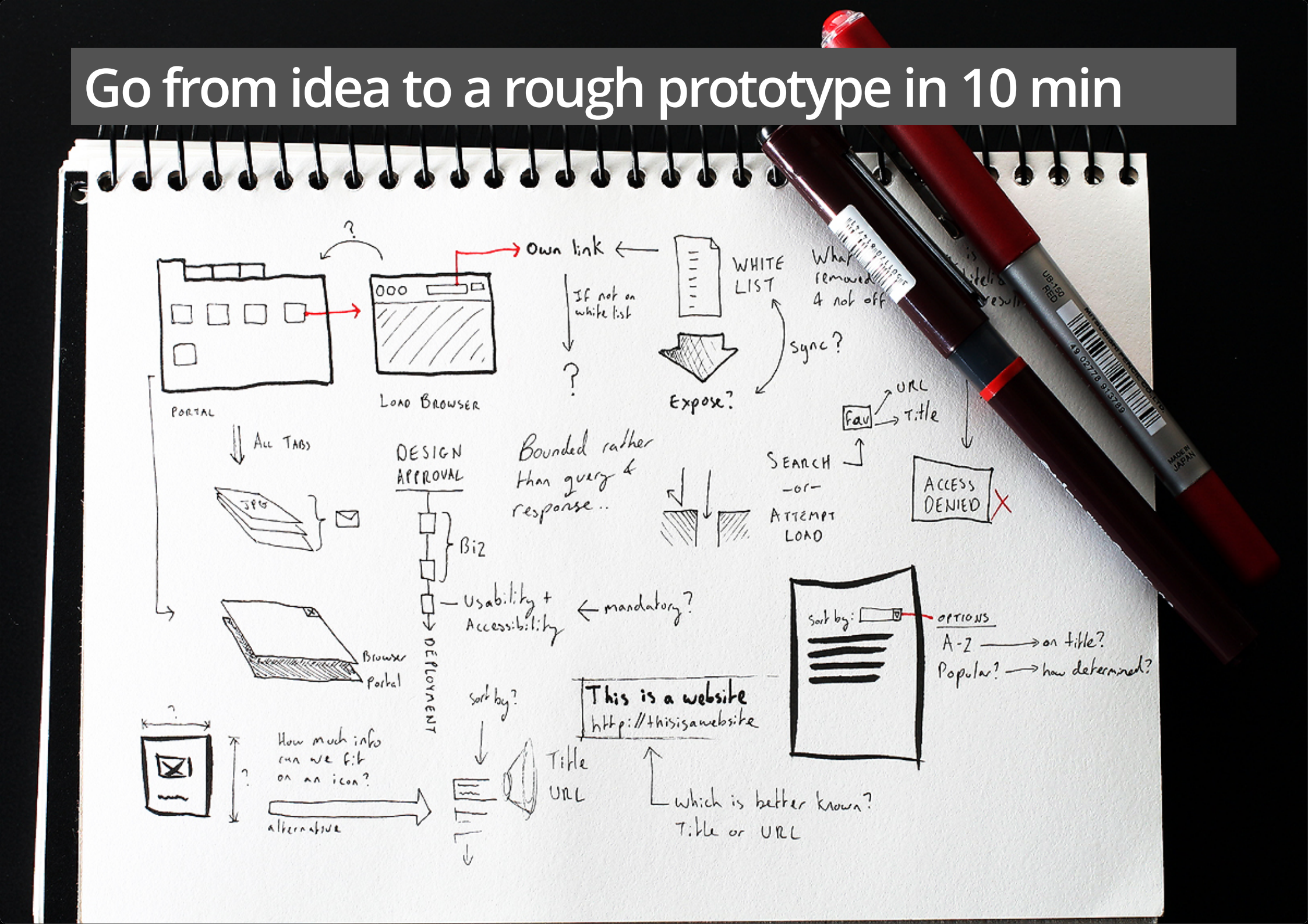
- Task based command line build tool
- Alternative to Rake/Cake/Make/Jake
- Rich community of build tasks
- Generates simple skeleton for new projects
- Lint, test, concat, watch and min out of the box.
- *however..you're still responsible for workflow*

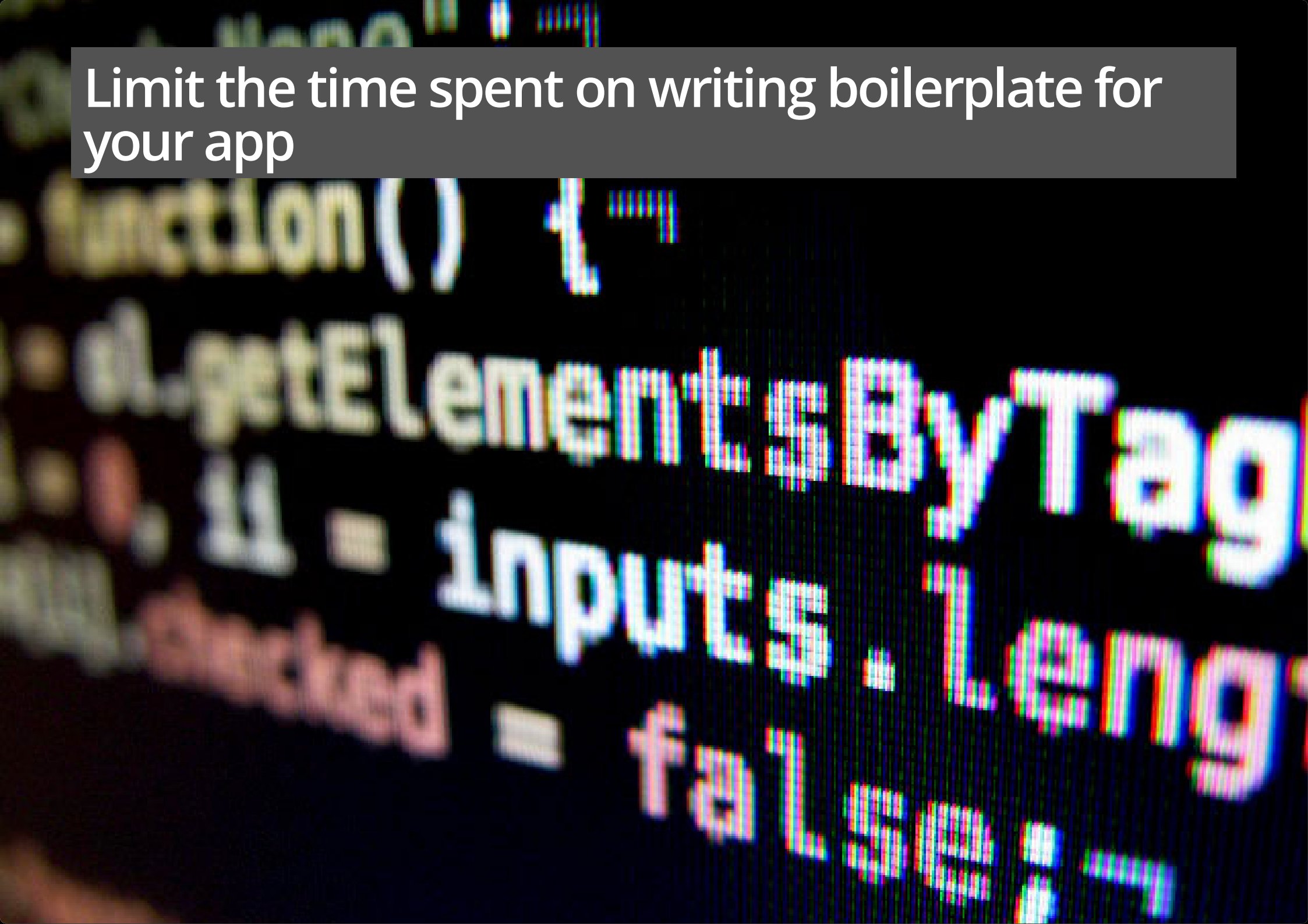


so much choice! you want flexibility.
how could we make this any easier?

Introducing Yeoman

Go from idea to a rough prototype in 10 min




Limit the time spent on writing boilerplate for your app


Automate as much of your workflow as possible

Yeoman

- Authoring abstractions
- Scaffolds
- Linting
- LiveReload
- Testing
- Build tools
- and more.

Built on top of great tools like Grunt and Bower

TWITTER ENGINEERING PRESENTS

BOWE

THE BROWSER PACKAGE MANAGER

HTML, CSS, AND JAVASCRIPT

GITHUB

Walkthrough

1. Package management
2. Generators
3. Live Reload
4. Testing
5. Build system

What's next?

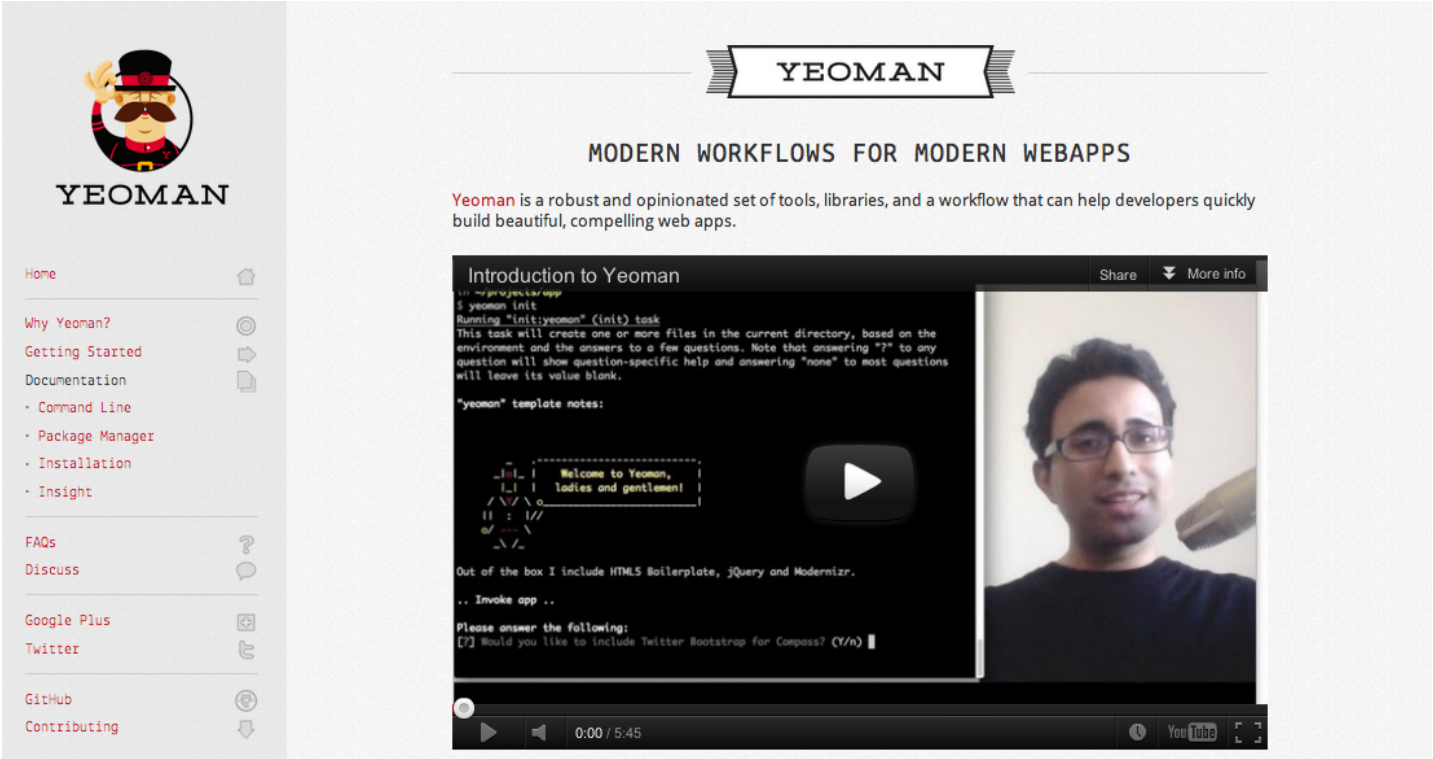

CHAPEL ROYAL OF
ST. PETER AD VINCULA
SUNDAY SERVICES
9-15-am. Holy Communion
11-00-am. Mattins & Sermon
E II R

Yeoman in 2013

1. Improved flexibility with tools (Grunt, Bower)
2. Better editor integration
3. Support for backends (Rails, PHP)
4. Better mobile helpers and remote debugging
5. Support for all of JS.next (Traceur)
6. Deployment (Heroku, AppEngine)
7. ..plans for a little more

Yeoman 0.9.5 Just Launched!

- Visit yeoman.io



Learn to love your workflow and tools

g+	addyosmani.com/+
twitter	@addyosmani
www	addyosmani.com
github	github.com/addyosmani

Thank you!